

[INICIAR SESIÓN](#)[NUESTROS PLANES](#)[TODOS LOS CURSOS](#)[FORMACIONES](#)[CURSOS](#)[PARA EMPRESAS](#)[ARTÍCULOS DE TECNOLOGÍA > PROGRAMACIÓN](#)

# Cómo separar palabras de String en Java



Alex Felipe

17/11/2020

Leí un archivo CSV con las informaciones nombre, correo electrónico y puntos de un alumno. Estas informaciones están contenidas en una única String:

```
String alumnoRecibido = "Alex; alex.vieira@caelum.com.br; 15000";
```

Ahora necesito enviarle estas informaciones:

```
Alumno alumno = new Alumno();  
alumno.setNombre (* ¡no puedo pasar toda la string! *);
```

## Capturando una parte de la String

¿Y ahora? ¿Cómo puedo conseguir una parte de esta String y settar en mi alumno?

Podemos usar el método `substring()`, de la clase `String`, que toma una parte usando 2 parámetros:

```
alumno.setNombre(alumnoRecibido.substring(0, 4));
```

```
System.out.println(alumno.getNombre());
```

Probando el código:

Alex

¡Genial, funcionó!

El método `substring()` capturó exactamente el nombre del alumno Alex, porque el nombre Alex comienza al principio (posición 0) y termina en el 4º carácter de esta String, es decir, para capturar el correo electrónico tendré que usar el `substring()` de nuevo. Entonces, veamos en qué posición está el correo electrónico...

El comienzo es 6, ahora el final ... 7, 8, 9, 10, 11, 12, 15, 17, 20 ... ¡Perdí la cuenta! ¡Qué tedioso! ¿Cada vez tendré que contar con los **dedos la posición del texto**?

¿Y si el alumno fuera Guillermo? No terminaría en la 4ª posición ... ¡Esto definitivamente será un gran problema!

## Separando palabras con el split

Además del método `substring()` la clase String tiene el método `split()` que **separa el texto en partes** según una regla, por ejemplo, quiero que mi texto se separe cada vez que aparezca un ";":

```
String[] textoSeparado = alumnoRecibido.split(";");
```

```
System.out.println(Arrays.toString(textoSeparado));
```

Viendo el array:

```
[Alex, alex.vieira@caelum.com.br, 15000]
```

¡Pudimos separar la información del alumno! Tenga en cuenta que el `split()` devolvió *un array* de String, pasa eso porque **separa cada línea en Strings diferentes**, por esta razón devuelve el array de String. Ahora podemos asignar cada valor a mi objeto alumno:

```
alumno.setNombre(textoSeparado[0]);  
alumno.setEmail(textoSeparado[1]);  
alumno.setPuntos(Integer.parseInt(textoSeparado[2]));
```

```
System.out.println(alumno.getNombre() + " - " + alumno.getEmail() + " - " + al
```

Verificación de las informaciones del alumno:

Exception in thread "main" java.lang.NumberFormatException: For input string:

## Precauciones al usar el split

Parece que nuestra conversión a `int` no funcionó... Apareció la *exception* porque en nuestra `String` tenemos un espacio (*backspace*) entre el ";" y las informaciones, es decir, en lugar de recibir sólo el número, se recibió el espacio junto con el número, por lo que no fue posible convertir. Si imprimieramos solo el correo electrónico, también vendría con el espacio:

```
System.out.println(alumno.getEmail());
```

Imprimiendo sólo el correo electrónico:

```
alex.vieira@caelum.com.br
```

## Aplicar expresión regular en el split

Necesitamos informar al `split()` para que elimine los espacios. El `split()` trabaja con [expresión regular](#), es decir, podemos especificar las reglas usando expresión regular. En este caso usaré la expresión "\s" que significa espacio en blanco:

```
String[] textoSeparado = alumnoRecibido.split(";\\s");
```

```
//setters
```

```
System.out.println(alumno.getNombre() + " - " + alumno.getEmail() + " - " + al
```

Ejecutando el código nuevamente:

```
Alex - alex.vieira@caelum.com.br - 15000
```

¡Funcionó! Sin embargo, hay un pequeño detalle... la expresión regular `";\s"` significa que **siempre** será cuando tenga ";" y un espacio juntos, es decir, si la String fuera:

```
String alumnoRecibido = "Alex;alex.vieira@caelum.com.br;15000";  
String[] textoSeparado = alumnoRecibido.split(";\\s");  
System.out.println(Arrays.toString(textoSeparado));
```

El resultado sería:

```
[Alex; alex.vieira@caelum.com.br;15000]
```

Seguramente mi sistema no funcionaría...

## Agregar más de una regla con pipe

Para hacer que el Split lidie con el caso de ";" o ";\\s" usamos el pipe (|):

```
String[] textoSeparado = alumnoRecibido.split(";|;\\s");
```

Volviendo a probar la misma String:

```
[Alex, alex.vieira@caelum.com.br, 15000]
```

¡Excelente!

Ahora podemos obtener todas las informaciones del alumno que provienen de la String. Es importante verificar que se utilizaron dos "\", ya que es necesario "escapar" de la barra para ser reconocida dentro de la String.

Vimos que cuando necesitamos separar texto en Java, tenemos el método `split()` de clase `String` que tiene la capacidad de separar el texto en partes según una regla especificada. En nuestro ejemplo se separó en todas las veces que aparecía ";" o cuando aparecían ";" más un espacio junto. También podría ser una "," o ":".

Lo importante es saber que el `split()` funciona con expresiones regulares, por lo que se aplica cualquiera de las dos.

En [Alura](#) tenemos toda una formación con varios cursos de [Java](#), desde lo más básico con el lenguaje hasta conceptos más avanzados como clases, poliformismo y herencia. Pero no te asustes con esos términos, ellos son abordados con un proyecto práctico que junto con nuestra metodología y didáctica hacen que el contenido sea mucho más fácil de ser aprendido.

Aprovecha nuestro descuento de lanzamiento y comienza a estudiar en Alura:

Plan trimestral de USD ~~29,90~~ por USD 19,90

<https://app.aluracursos.com/compra/trimestral/promo-lanzamiento>

ARTÍCULOS DE TECNOLOGÍA > PROGRAMACIÓN

**En Alura encontrarás variados cursos sobre Programación.  
¡Comienza ahora!**

**SEMESTRAL**

**US\$49,90**

un solo pago de US\$49,90

✓ 218 cursos

- ✓ Videos y actividades 100% en Español
- ✓ Certificado de participación
- ✓ Estudia las 24 horas, los 7 días de la semana
- ✓ Foro y comunidad exclusiva para resolver tus dudas
- ✓ Acceso a todo el contenido de la plataforma por 6 meses

**¡QUIERO EMPEZAR A ESTUDIAR!**

[Paga en moneda local en los siguientes países](#)

**ANUAL**

**US\$79,90**

un solo pago de US\$79,90

- ✓ 218 cursos
- ✓ Videos y actividades 100% en Español

- ✓ Certificado de participación
- ✓ Estudia las 24 horas, los 7 días de la semana
- ✓ Foro y comunidad exclusiva para resolver tus dudas
- ✓ Acceso a todo el contenido de la plataforma por 12 meses

**¡QUIERO EMPEZAR A ESTUDIAR!**

[Paga en moneda local en los siguientes países](#)

Acceso a todos  
los cursos

Estudia las 24 horas,  
dónde y cuándo quieras

Nuevos cursos  
cada semana

## NAVEGACIÓN

PLANES  
INSTRUCTORES  
BLOG  
POLÍTICA DE PRIVACIDAD  
TÉRMINOS DE USO  
SOBRE NOSOTROS  
PREGUNTAS FRECUENTES

## ¡CONTÁCTANOS!

¡QUIERO ENTRAR EN CONTACTO!

## BLOG

PROGRAMACIÓN  
FRONT END  
DATA SCIENCE  
INNOVACIÓN Y GESTIÓN  
DEVOPS

AOVS Sistemas de Informática S.A  
CNPJ 05.555.382/0001-33

## SÍGUENOS EN NUESTRAS REDES SOCIALES



## ALIADOS





En Alura somos unas de las Scale-Ups seleccionadas por Endeavor, programa de aceleración de las empresas que más crecen en el país.



Fuimos unas de las 7 startups seleccionadas por Google For Startups en participar del programa Growth Academy en 2021

POWERED BY

## CURSOS

### Cursos de Programación

Lógica de Programación | Java

### Cursos de Front End

HTML y CSS | JavaScript | React

### Cursos de Data Science

Data Science | Machine Learning | Excel | Base de Datos | Data Visualization | Estadística

### Cursos de DevOps

Docker | Linux

### Cursos de Innovación y Gestión

Productividad y Calidad de Vida | Transformación Ágil | Marketing Analytics | Liderazgo y Gestión de Equipos | Startups y Emprendimiento